Approximate parametric cone programming with applications in control

Wannes Van Loock, Erik Lambrechts, Gijs Hilhorst, and Goele Pipeleers

Abstract— **Parametric programming analyzes the solution of parameter-dependent optimization problems as a function of the parameters. As the true parametrized solution is generally too complicated to be practicable in applications, research has turned to computing adequate approximate solutions. This paper presents a novel approach to compute such approximations for parametric cone programs with a polynomial parameter dependency. A piecewise polynomial parametrization is adopted for the optimizer function, and the coefficients are optimized to minimize the average suboptimality over the parameter domain. The resulting semi-infinite optimization problem is transformed into a tractable, yet conservative optimization problem by exploiting the positivity of the B-spline basis functions. Relying on duality, bounds on the suboptimality of the approximation are computed which can be used to locally refine the solution. The approach is implemented in an open source software tool and illustrated by three applications in control.**

I. Introduction

Parametric programming considers optimization problems in which the problem data are affected by one or more parameters, and aims at describing the optimal value and an optimizer as explicit functions of the parameters (see e.g. [1] and references therein). It traces back as far as the 50s for as soon as people were enabled to solve real decision problems by Dantzig's simplex method, they realized the dependency of the outcome on the numerical problem data. Spurred by a broad range of applications, research on parametric programming has steadily grown over the last decades. For instance, computing the Pareto front of a multi-objective optimization problem amounts to solving a parametric program. In addition, parametric programming has found application in model predictive control (MPC), a control strategy that solves an optimization problem at every time sample to compute the next control action. These on line optimization problems only differ in the current sensor measurements, which affect the problem data. So-called explicit MPC solves the corresponding parametric program off line, yielding the optimal controls as an explicit function of the sensor measurements [1], [2]. Furthermore, parametric programming has been proven valuable in solving bi-level optimization problems [3].

Acknowledgement: This work benefits from IWT SBO project MBSE4Mechatronics: Model-based Systems Engineering for Mechatronics, from KU Leuven-BOF PFV/10/002 Center-of-Excellence Optimization in Engineering (OPTEC) and from the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office (DYSCO).

The authors are with the Department of Mechanical Engineering, Division PMA, KU Leuven, 3001 Leuven, Belgium. goele.pipeleers@kuleuven.be

Initially, research in parametric programming focused on characterizing the true solution to the problem. This generally comes down to determining so-called critical or characteristic regions: sets of parameter values for which the solution of the optimization problem features similar properties (e.g. a specific active set), and explicitly characterizing the optimal value and an optimizer as a function of the parameters within each of these regions. Unfortunately, the number of regions grows exponentially with the number of parameters and optimization variables, and in most cases the solution within a region is not expressible in analytical form [1]. As these disadvantages severely compromise the practical applicability of the result, researchers got attracted to the construction of approximate, yet easy-to-evaluate solutions to parametric programming problems [4]–[10]. The vast majority of these constructions apply to parametric convex programs in which the parameters only affect the bounds (right-hand sides) of the constraints.

In deriving an approximate solution, one generally proposes a certain parametrization for the optimizer function (e.g. polynomial) and optimizes the corresponding coefficients. In doing so, the optimizer function must be guaranteed to return a feasible solution to the parametric program for all parameter values. This guarantee is generally provided by exploiting the convexity of the problem [5], [6], [9] or by using Polya's [7] or sum-of-squares [8], [10] relaxations. In addition, bounds on the suboptimality of the approximate solution are generally derived, based on which the proposed parametrization can be refined where needed.

This paper presents a novel approach to constructing approximate solutions to parametric convex cone problems. Although following a similar pattern, it differs from the aforementioned approaches in the following respects: (i) it applies to a large class of parametric programs in which the parameters are allowed to affect the constraint functions as well as the bounds; (ii) a piecewise polynomial parametrization of the optimizer function is proposed with direct control over the overall continuity and smoothness; (iii) feasibility for all parameter values is guaranteed by exploiting the positivity of the chosen basis functions; and (iv) tight bounds on the suboptimality of the approximation are computed using duality. The approach is implemented in an open source software tool [11], and illustrated by numerical examples in control: the computation of a Pareto front between conflicting control requirements; the approximation of uncertainty sets to render them amenable to robust control synthesis; and a bilevel approach for solving bilinear matrix inequalities in structured control design.

The paper is organized as follows: After presenting the considered class of parametric programs, Section III describes the proposed procedure for constructing approximate solutions. Section IV extends the approach to handle infeasible parameter values, and Section V illustrates the approach by numerical examples. The notation used is standard. Bold capital letters are used to indicate sets, and for a set $\mathbf{A}$, $|\mathbf{A}|$ denotes its number of elements.

II. Problem statement

Consider the following parametric cone program

$$\begin{aligned} \Pi(\theta) \quad : \quad & \underset{x \in \mathbf{R}^n}{\text{minimize}} && c(\theta)^\intercal x \\ & \text{subject to} && \mathcal{A}(\theta)x + b(\theta) \preceq_{\mathbf{K}} 0, \end{aligned} \tag{1}$$

where $\theta \in \mathbf{\Theta} \subseteq \mathbf{R}^t$ are the parameters affecting the problem data. For every θ, $\mathcal{A}(\theta)$ is a linear mapping from $\mathbf{R}^n$ to $\mathbf{Y}$, which corresponds to the Cartesian product of Euclidean spaces and/or vector spaces of symmetric matrices. The cone $\mathbf{K} \subset \mathbf{Y}$ is a direct product of nonnegative orthants, second order cones and/or semidefinite cones.

The set of all parameter values for which $\Pi(\theta)$ is feasible, respectively strictly feasible, is denoted by $\mathbf{\Theta}_{\Pi_\mathrm{f}}$, respectively $\mathbf{\Theta}_{\Pi_\mathrm{sf}}$:

$$\begin{aligned} \mathbf{\Theta}_{\Pi_\mathrm{f}} &= \left\{ \theta \in \mathbf{R}^t \;:\; \exists x,\ \mathcal{A}(\theta)x + b(\theta) \preceq_{\mathbf{K}} 0 \right\}, \\ \mathbf{\Theta}_{\Pi_\mathrm{sf}} &= \left\{ \theta \in \mathbf{R}^t \;:\; \exists x,\ \mathcal{A}(\theta)x + b(\theta) \prec_{\mathbf{K}} 0 \right\}. \end{aligned}$$

The *optimal value function* $p^\star : \mathbf{\Theta}_{\Pi_\mathrm{f}} \to \mathbf{R}$ associates with every θ the optimal value of $\Pi(\theta)$. An *optimizer function* $x^\star : \mathbf{\Theta}_{\Pi_\mathrm{f}} \to \mathbf{R}^n$ associates with each parameter $\theta \in \mathbf{\Theta}_{\Pi_\mathrm{f}}$ an optimizer $x^\star(\theta)$.

As the considered cones $\mathbf{K}$ are self-dual, the dual cone program of $\Pi(\theta)$ amounts to

$$\begin{aligned} \Delta(\theta) \quad : \quad & \underset{y \in \mathbf{Y}}{\text{maximize}} && \langle b(\theta), y \rangle \\ & \text{subject to} && \mathcal{A}(\theta)^* y + c(\theta) = 0 \\ & && y \succeq_{\mathbf{K}} 0, \end{aligned} \tag{2}$$

where $\mathcal{A}(\theta)^* : \mathbf{Y} \to \mathbf{R}^n$ denotes the adjoint mapping of $\mathcal{A}(\theta)$. The set of all parameter values for which $\Delta(\theta)$ is (strictly) feasible is denoted by ($\mathbf{\Theta}_{\Delta_\mathrm{sf}}$) $\mathbf{\Theta}_{\Delta_\mathrm{f}}$. The dual optimal value function is indicated by $d^\star$, and dual optimizer functions are denoted by $y^\star$.

Initially, we will rely on the assumptions below. Assumption 3 will be relaxed in Section IV.

A1. The optimization data c, b and $\mathcal{A}$ depend polynomially on θ.
A2. The considered set $\mathbf{\Theta}$ is a hyperrectangle in $\mathbf{R}^t$.
A3. For all $\theta \in \mathbf{\Theta}$, $\Pi(\theta)$ and $\Delta(\theta)$ are strictly feasible.

The last assumption guarantees that for all $\theta \in \mathbf{\Theta}$ both $p^\star(\theta)$ and $d^\star(\theta)$ are finite, and strong duality holds:

$$-\infty < p^\star(\theta) = d^\star(\theta) < \infty .$$

In addition, this assumption implies that both $p^\star(\theta)$ and $d^\star(\theta)$ are attained, such that optimizer functions $x^\star$ and $y^\star$ are guaranteed to exist.

III. B-spline parametrized solutions

In this section we present an approach for computing piecewise polynomial approximate optimizer function to $\Pi(\theta)$ and $\Delta(\theta)$ for $\theta \in \mathbf{\Theta}$. The approximate optimizer function, indicated by $\hat{x}$ for the primal problem, is such that it (i) yields a feasible point $\hat{x}(\theta)$ of $\Pi(\theta)$ for all $\theta \in \mathbf{\Theta}$; and (ii) minimizes the average suboptimality over $\mathbf{\Theta}$. For the sake of conciseness, we focus on the primal problem below. The derivation of $\hat{y}$ for the dual problem is completely analogous.

We propose to parametrize the optimizer function as a linear combination of preset basis functions $b^x : \mathbf{\Theta} \to \mathbf{R}$, labeled by the (multi-)index α:

$$\hat{x}(\mathsf{x}) = \sum_{\alpha \in \boldsymbol{\alpha}_{b^x}} \mathsf{x}_\alpha \, b^x_\alpha , \tag{3}$$

where $\mathsf{x} \in \mathbf{R}^{n|\boldsymbol{\alpha}_{b^x}|}$. The image of $\hat{x}(\mathsf{x})$ for a particular θ is indicated by $\hat{x}(\mathsf{x}, \theta)$.

By minimizing the average suboptimality over $\mathbf{\Theta}$, we arrive at the following semi-infinite optimization problem

$$\begin{aligned} \hat{\Pi} \quad : \quad & \underset{\mathsf{x} \in \mathbf{R}^{n|\boldsymbol{\alpha}_{b^x}|}}{\text{minimize}} && \int_{\mathbf{\Theta}} c(\theta)^\intercal \hat{x}(\mathsf{x}, \theta) \, \mathrm{d}\theta \\ & \text{subject to} && \mathcal{A}(\theta)\hat{x}(\mathsf{x}, \theta) + b(\theta) \preceq_{\mathbf{K}} 0, \\ & && \qquad\qquad \forall \theta \in \mathbf{\Theta}. \end{aligned} \tag{4}$$

To convert $\hat{\Pi}$ to a tractable optimization problem, a tensor product B-spline parametrization is adopted for $\hat{x}$. In this way the semi-infinite constraints can be relaxed, that is: replaced by a more restrictive finite sets of constraints, by exploiting the positivity of the basis functions.

A. Tensor product B-splines

Let us first consider univariate piecewise polynomials on a finite closed interval $\mathbf{\Theta} = [\underline{\theta}, \bar{\theta}] \subset \mathbf{R}$ that is partitioned as

$$\underline{\theta} = \xi_0 < \xi_1 < \cdots < \xi_l < \xi_{l+1} = \bar{\theta} .$$

Let us consider the vector space of piecewise polynomials whose restriction to $[\xi_i, \xi_{i+1}]$, $i = 0, \ldots, l$, is a polynomial of order k and satisfy given continuity conditions at the break points ξ_i, $i = i, \ldots, l$. By the Curry Schoenberg theorem (see e.g. [12], [13]), one can always construct a knot sequence λ such that every element s in this space can be represented uniquely as a linear combination of B-splines $b_{i,k,\lambda}$ of order k with knot sequence λ:

$$s = \sum_{i=1}^{|\lambda|-k} \mathsf{s}_i b_{i,k,\lambda} .$$

The coefficients s_i are called the B-spline coefficients of s. The B-spline basis exhibits a number of useful properties:

1. Positivity: $b_{i,k,\lambda}(\theta) \geq 0$, for all $\theta \in \mathbf{R}$.
2. Local support: $b_{i,k,\lambda}(\theta) = 0$, for all $\theta \notin [\lambda_i, \lambda_{i+k}]$.
3. Partition of unity: $\sum b_{i,k,\lambda}(\theta) = 1$, for all $\theta \in [\underline{\theta}, \bar{\theta}]$.

Tensor product splines constitute a particular multivariate generalization of univariate splines using the tensor product construct. To define tensor product splines on

$$\mathbf{\Theta} = [\underline{\theta}_1, \bar{\theta}_1] \times \cdots \times [\underline{\theta}_t, \bar{\theta}_t] \subset \mathbf{R}^t ,$$

a degree k_j and knot sequence λ_j, $j = 1, \ldots, t$ is chosen for every coordinate θ_j. Let the corresponding B-spline basis for every coordinate be indicated by b_{k_j,λ_j}, then the corresponding tensor product B-splines are defined as

$$b_{\alpha,k,\lambda}(\theta) = \prod_{j=1}^{t} b_{\alpha_j,k_j,\lambda_j}(\theta_j),$$

for $\alpha \in \boldsymbol{\alpha}$:

$$\boldsymbol{\alpha} = \{1, \ldots, |\lambda_1| - k_1\} \times \cdots \times \{1, \ldots, |\lambda_t| - k_t\}.$$

Every linear combination of $b_{\alpha,k,\lambda}$, $\alpha \in \boldsymbol{\alpha}$, is called a tensor product spline. The properties of the coordinate B-splines transfer to the tensor product B-splines $b_{\alpha,k,\lambda}$: the basis functions are linearly independent, positive, sum up to one (partition of unity), and have local support. Also, note that the restriction of a multivariate polynomial to $\boldsymbol{\Theta}$ is a tensor product spline.

B. Tractable reformulation

Let us substitute a tensor product B-spline basis for b^x in (3). Then due to Assumption 1, the constraint function

$$f(\mathsf{x}, \theta) = \mathcal{A}(\theta)\hat{x}(\mathsf{x}, \theta) + b(\theta)$$

is a tensor product spline in θ. Let b^f denote the corresponding tensor-product B-spline basis, with its entries labeled by $\alpha \in \boldsymbol{\alpha}_{b^f}$. Hence, f can be written in the form

$$f(\mathsf{x}, \theta) = \sum_{\alpha \in \boldsymbol{\alpha}_{b^f}} \mathsf{f}_\alpha(\mathsf{x}) b^f_\alpha(\theta).$$

The coefficients f are readily verified to be affine in x as the constraint function is affine in $\hat{x}$, which in turn depends linearly on x.

Now consider the following optimization problem

$$\begin{aligned} \hat{\Pi}_{\text{rel}} \;:\quad & \underset{\mathsf{x} \in \mathbf{R}^{n|\boldsymbol{\alpha}_{b^x}|}}{\text{minimize}} && \int_{\boldsymbol{\Theta}} c(\theta)^\intercal \hat{x}(\mathsf{x}, \theta)\, \mathrm{d}\theta \\ & \text{subject to} && \mathsf{f}_\alpha(\mathsf{x}) \preceq_{\mathbf{K}} 0, \quad \forall \alpha \in \boldsymbol{\alpha}_{b^f}. \end{aligned} \tag{5}$$

Due to the positivity of tensor-product B-spline bases every feasible point of (5) is feasible for (4). Consequently, every solution $\mathsf{x}^\star$ of (5) provides an upper bound to the optimal value function $p^\star$ on $\boldsymbol{\Theta}$:

$$p^\star(\theta) \le c(\theta)^\intercal \hat{x}(\mathsf{x}^\star, \theta), \quad \forall \theta \in \boldsymbol{\Theta}. \tag{6}$$

C. Duality

In order to assess the suboptimality of the approximate optimizer function $\hat{x}(\mathsf{x}^\star)$, an approximate solution to the dual problem (2) is computed similarly as explained above for the primal. That is: a tensor product spline approximate dual optimizer function is proposed

$$\hat{y}(\mathsf{y}) = \sum_{\alpha \in \boldsymbol{\alpha}_{b^y}} \mathsf{y}_\alpha\, b^y_\alpha, \tag{7}$$

and the coefficients y are computed as the solution of the following optimization problem:

$$\begin{aligned} \hat{\Delta}_{\text{rel}} \;:\quad & \underset{\mathsf{y} \in \mathbf{Y}^{|\boldsymbol{\alpha}_{b^y}|}}{\text{minimize}} && \int_{\boldsymbol{\Theta}} \langle b(\theta), \hat{y}(\mathsf{y}, \theta) \rangle\, \mathrm{d}\theta \\ & \text{subject to} && \mathsf{g}_\alpha(\mathsf{y}) = 0, \quad \forall \alpha \in \boldsymbol{\alpha}_{b^g} \\ & && \mathsf{y}_\alpha \succeq_{\mathbf{K}} 0, \quad \forall \alpha \in \boldsymbol{\alpha}_{b^y}. \end{aligned} \tag{8}$$

The linear functions $\mathsf{g}(\mathsf{y})$ return the coefficients of

$$g(\mathsf{y}, \theta) = \mathcal{A}(\theta)^* \hat{y}(\mathsf{y}, \theta) + c(\theta)$$

in the corresponding tensor product B-spline basis. By the linear independence of tensor product B-splines, the equalities in (8) are equivalent to

$$g(\mathsf{y}, \theta) = 0, \quad \forall \theta \in \boldsymbol{\Theta}. \tag{9}$$

Note that care must be taken in selecting the parametrization (7) such that this equality admits a solution. By the positivity of tensor product B-splines, the inequalities in (8) imply

$$\hat{y}(\mathsf{y}, \theta) \succeq_{\mathbf{K}} 0, \quad \forall \theta \in \boldsymbol{\Theta}.$$

Consequently, for every solution $\mathsf{y}^\star$ of (8), $\hat{y}(\mathsf{y}^\star, \theta)$ is feasible for (2) for all $\theta \in \boldsymbol{\Theta}$ such that

$$\langle b(\theta), \hat{y}(\mathsf{y}^\star, \theta) \rangle \le d^\star(\theta), \quad \forall \theta \in \boldsymbol{\Theta}. \tag{10}$$

Combining (6) and (10) we get both an upper and lower bound on the true optimizer function $d^\star(\theta) = p^\star(\theta)$.

D. Refining the Relaxations

The distance between the upper bound (6) and lower bound (10) has a twofold origin: (i) the particular parametrization adopted for $\hat{x}$ and $\hat{y}$; and (ii) the conservatism introduced by relaxing semi-infinite constraints of the form

$$s(\theta) \preceq_{\mathbf{K}} 0, \quad \forall \theta \in \boldsymbol{\Theta}$$

with s a tensor product spline, to finite sets of constraints

$$\mathsf{s}_\alpha \preceq_{\mathbf{K}} 0, \quad \forall \alpha \in \boldsymbol{\alpha}_{b^s} \tag{11}$$

with s_α the coefficients of s in the tensor product B-spline basis b^s. To mitigate the latter cause, s is represented in a higher-dimensional basis $\tilde{b}^s$

$$\operatorname{span}\{b^s\} \subset \operatorname{span}\{\tilde{b}^s\}$$

before adopting the relaxation. That is, the coefficients $\tilde{\mathsf{s}}_\alpha$ of s in the extended basis $\tilde{b}^s$ are computed

$$s = \sum_{\alpha \in \boldsymbol{\alpha}_{b^s}} \mathsf{s}_\alpha b^s_\alpha = \sum_{\alpha \in \boldsymbol{\alpha}_{\tilde{b}^s}} \tilde{\mathsf{s}}_\alpha \tilde{b}^s_\alpha\,,$$

and (11) is replaced by the larger, yet less conservative set of constraints

$$\tilde{\mathsf{s}}_\alpha \preceq_{\mathbf{K}} 0, \quad \forall \alpha \in \boldsymbol{\alpha}_{\tilde{b}^s}. \tag{12}$$

Two ways of constructing such refinements have been reported in the spline literature (see e.g. [12], [13]): one is to elevate the degree of the spline basis, the other is to insert additional knots in the spline basis. Both approaches are guaranteed to be asymptotically exact. The former yields a global reduction of the conservatism over the entire domain $\boldsymbol{\Theta}$, while the latter allows for local reductions, which are generally larger.

IV. Infeasible parameter values

To relax Assumption 3 a two-step procedure is proposed. Below we consider the case where $\boldsymbol{\Theta} \subseteq \boldsymbol{\Theta}_{\Delta_{\mathrm{f}}}$ but $\boldsymbol{\Theta} \nsubseteq \boldsymbol{\Theta}_{\Pi_{\mathrm{f}}}$; the procedure for $\boldsymbol{\Theta} \subseteq \boldsymbol{\Theta}_{\Pi_{\mathrm{f}}}$ but $\boldsymbol{\Theta} \nsubseteq \boldsymbol{\Theta}_{\Delta_{\mathrm{f}}}$ is completely analogous. In the first step, we approximately solve the following parametric phase I problem for $\Pi(\theta)$:

$$\begin{aligned}\Phi(\theta) \ : \quad & \underset{x\in\mathbf{R}^n, t\in\mathbf{R}}{\text{minimize}} \quad t \\ & \text{subject to} \quad \mathcal{A}(\theta)x + b(\theta) \preceq_{\mathbf{K}} \mathbf{1}t \ ,\end{aligned}$$

as well as the corresponding dual

$$\begin{aligned}\Psi(\theta) \ : \quad & \underset{y\in\mathbf{Y}}{\text{maximize}} \quad \langle b(\theta), y\rangle \\ & \text{subject to} \quad \mathcal{A}(\theta)^* y = 0 \\ & \qquad\qquad\quad \langle \mathbf{1}, y\rangle = 1 \\ & \qquad\qquad\quad y \succeq_{\mathbf{K}} 0 \ .\end{aligned}$$

Note that both $\Phi(\theta)$ and $\Psi(\theta)$ are strictly feasible for all parameter values $\theta \in \boldsymbol{\Theta}$ so that we can apply the methodology of Section III to get approximate optimizer functions $\hat{x}$, $\hat{t}$ and $\hat{y}$. These allow us to determine $\boldsymbol{\Theta}_{\Pi_{\mathrm{f}}} \subset \boldsymbol{\Theta}$ as

$$\begin{aligned}\hat{t}(\theta) \leq 0 \quad &\Rightarrow \quad \theta \in \boldsymbol{\Theta}_{\Pi_{\mathrm{f}}} \ , \\ \langle b(\theta), \hat{y}(\theta)\rangle > 0 \quad &\Rightarrow \quad \theta \notin \boldsymbol{\Theta}_{\Pi_{\mathrm{f}}} \ .\end{aligned}$$

In the second step, we apply the developed methodology to approximately solve the relaxed problem

$$\begin{aligned}\Pi_{\hat{t}_+}(\theta) \ : \quad & \underset{x\in\mathbf{R}^n}{\text{minimize}} \quad c(\theta)^\intercal x \\ & \text{subject to} \quad \mathcal{A}(\theta)x + b(\theta) \preceq_{\mathbf{K}} \mathbf{1}\hat{t}_+(\theta) \ ,\end{aligned}$$

where $\hat{t}_+$ is constructed from $\hat{t}$ by replacing its negative tensor product B-spline coefficients by zero such that

$$\hat{t}_+(\theta) \gtrapprox \max\{0, \hat{t}(\theta)\} \ , \quad \forall \theta \in \boldsymbol{\Theta} \, .$$

Note that $\hat{t}_+(\theta)$ would be the result of our approach applied to $\Phi(\theta)$ supplemented with the constraint $t \geq 0$. The dual problem of $\Pi_{\hat{t}_+}(\theta)$ is given by

$$\begin{aligned}\Delta_{\hat{t}_+}(\theta) : \quad & \underset{y\in\mathbf{Y}}{\text{maximize}} \quad \langle b(\theta) - \hat{t}_+(\theta), y\rangle \\ & \text{subject to} \quad \mathcal{A}(\theta)^* y + h(\theta) = 0 \\ & \qquad\qquad\quad y \succeq_{\mathbf{K}} 0 \ .\end{aligned}$$

V. Applications

Parametric programming has a broad range of applications. Below, we illustrate the developed approach for computing approximate parametric programming solutions in three applications related to control: the computation of Pareto fronts between conflicting control requirements (Section V-A); the approximation of uncertainty sets to render them amenable to robust control synthesis (Section V-B); and a bilevel approach for solving bilinear matrix inequalities in structured control design (Section V-C). To facilitate the implementation of optimization problems containing splines a MATLAB software toolbox has been developed and made available at Gitlab [11]. It is based on YALMIP [14] and facilitates the formulation of optimization problems with scalar, vector or matrix valued splines. The relaxation of the semi-infinite problem is performed behind the scenes. The presented examples are shipped with the source code to allow for future benchmarking.

A. Trade-off analysis

To illustrate the methodology, we apply it to efficiently compute an approximation of a trade-off curve considered in [15]. This paper considers the design of repetitive controllers, which are bound to a trade-off between the attenuation of periodic disturbances and the amplification of nonperiodic disturbances. Both objectives are quantified in a performance index, denoted by γ_{p}, respectively, γ_{np}, and the corresponding Pareto front is computed by solving the following semi-definite program for various values of γ_{np}:

$$\begin{aligned}& \underset{C,P,Q,R,\gamma_{\mathrm{p}}}{\text{minimize}} \quad \gamma_{\mathrm{p}} \\ & \text{subject to} \quad Z_{\mathrm{np}}(P, C, \gamma_{\mathrm{np}}) \preceq 0 \\ & \qquad\qquad\quad Z_{\mathrm{p}}(Q, R, C, \gamma_{\mathrm{p}}) \preceq 0 \\ & \qquad\qquad\quad R \succeq 0 \, ,\end{aligned} \tag{13}$$

where $Z_{\mathrm{np}}(P, C, \gamma_{\mathrm{np}})$ is given by

$$\begin{bmatrix} A^\intercal PA - P & A^\intercal PB & C^\intercal \\ B^\intercal PA & B^\intercal PB - \gamma_{\mathrm{np}} & D^\intercal \\ C & D & -\gamma_{\mathrm{np}} \end{bmatrix}$$

and $Z_{\mathrm{p}}(Q, R, C, \gamma_{\mathrm{p}})$ by

$$\begin{bmatrix} Z_{11} & (A^\intercal Q + R)B & C^\intercal \\ B^\intercal(QA + R) & B^\intercal QB - \gamma_{\mathrm{p}} & D^\intercal \\ C & D & -\gamma_{\mathrm{p}} \end{bmatrix}$$

with

$$Z_{11} = A^\intercal QA - Q + RA + A^\intercal R + \eta R \, .$$

Details on how to set A, B, D and η are found in [15]. Fig. 1 shows the corresponding Pareto front for a particular repetitive controller design. The black line indicates the true trade-off curve obtained by griding, while the gray lines indicate approximations obtained with the proposed approach applied to (13) with $\theta = \gamma_{\mathrm{np}}$ as parameter. Approximate solutions to the primal provide upper bounds to the Pareto front; approximate solutions to the dual lower bounds. Initially, rough approximations using cubic splines with only four internal knots are considered. The results delineate the dark gray shaded area in Fig. 1, and are clearly suboptimal. As pointed out in Section III-D, this suboptimality has a twofold origin. One source is the conservatism of the constraint relaxations, and if we reduce this as outlined in Section III-D, the suboptimality reduces from the dark gray shaded area to the medium gray shaded area. The remaining suboptimality is due to the chosen parametrization of the optimizer functions. After refining the parametrization by including additional knots at 1.2, 1.4, 2.4 and 2.6, it reduces to the light gray shaded area.

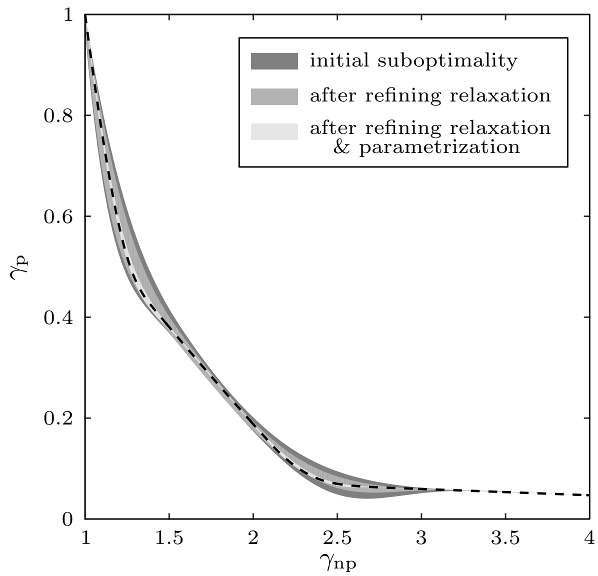

Fig. 1. Trade-off curve between γ_{np} and γ_{p} (black dashed), and three approximations.

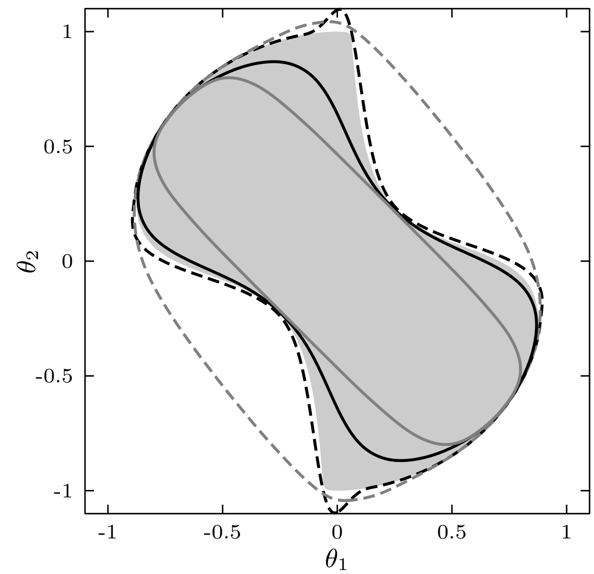

Fig. 2. Nonconvex (black) and convex (gray) inner (solid) and outer (dashed) approximations of the set $\mathbf{P}$ (shaded).

B. Approximation of polynomial matrix inequalities

We consider the problem of determining (convex) inner and outer spline approximations of the set defined by $\mathbf{P} = \{\theta \in \mathbf{R}^t \ : \ P(\theta) \preceq_{\mathbf{K}} 0\}$ where $P \ : \ \mathbf{R}^t \to \mathbf{Y}$ depends polynomially on θ. Such sets are common in stability analysis of linear systems where stability can be formulated semialgebraically in the space of coefficients of the characteristic polynomial. Using the Hermite stability criterion, these problems can be formulated as polynomial matrix inequalities, which typically are non-convex [16]. Subsequently, these approximations can be used to synthesize (robust) controllers over a simpler scalar (and convex) description of the feasible set.

By determining the approximate optimizer function, $\hat{x}$, to the parametric phase I feasibility problem

$$\begin{array}{ll} \underset{x\in\mathbf{R}}{\text{minimize}} & x \\ \text{subject to} & P(\theta) \preceq_{\mathbf{K}} xI, \end{array}$$

an inner approximation is given by

$$\underline{\mathbf{P}} = \{\theta \in \mathbf{R}^t \ : \ \hat{x}(\theta) \leq 0\}.$$

Similarly, the approximate optimizer function, $\hat{y}$, to the dual

$$\begin{array}{ll} \underset{y\in\mathbf{Y}}{\text{maximize}} & \mathrm{Tr}(yP(\theta)) \\ \text{subject to} & y \succeq_{\mathbf{K}} 0, \\ & \mathrm{Tr}(y) = 1 \end{array}$$

yields the outer approximation

$$\bar{\mathbf{P}} = \{\theta \in \mathbf{R}^t \ : \ \mathrm{Tr}(\hat{y}(\theta)P(\theta)) \geq 0\}.$$

Convexity of the inner (outer) approximations can easily be imposed by constraining the Hessian of the objective function to be positive (negative) semidefinite.

To illustrate the idea, consider the set $\mathbf{P}$ from [16] defined by the polynomial matrix inequality

$$P(\theta) = \begin{pmatrix} 1 - 16\theta_1\theta_2 & \theta_1 \\ \theta_1 & 1 - \theta_1^2 - \theta_2^2 \end{pmatrix} \succeq 0.$$

Fig. 2 shows the nonconvex (black) and convex (gray) inner and outer approximations of the set $\mathbf{P}$, indicated in dark gray. A degree 3 B-spline basis with 13 internal knots was chosen in both coordinates, both for the primal and the dual solution.

C. Bilinear programming

Consider the following bilinear program

$$\begin{array}{ll} \underset{x\in\mathbf{R}^n, z\in\mathbf{R}^m}{\text{minimize}} & c(z)^{\mathsf{T}}x + d(z) \\ \text{subject to} & \mathcal{A}(z)x + b(z) \preceq_{\mathbf{K}} 0. \end{array} \tag{14}$$

The above problem can be regarded as a parametric program with variables x and parameters $\theta = z$. An approximate solution to (14) is determined by first determining an approximate solution $\hat{x}(z)$ and subsequently doing an unconstrained minimization over z

$$\underset{z\in\mathbf{R}^m}{\text{minimize}} \quad c(z)^{\mathsf{T}}\hat{x}(z) + d(z).$$

Although this minimization is generally non-convex, it is smooth and differentiable and an accurate initial guess for the solver follows from the location of the control point corresponding to the minimum value of the B-spline coefficients [12], [13]. The solution to this problem provides an upper bound on the optimal value. Similarly, the minimization of the approximate dual solution of (14) provides a lower bound.

Problem	Bilevel parametric	SCP [17]	HIFOO
AC7	0.032 07	0.0339	0.0651
AC17	6.6124	6.6571	6.6124
HE1	0.174 53	0.2188	0.1540
EB1	1.8985	2.0532	3.1225
EB2	0.814 24	0.8150	2.0201
EB3	0.814 25	0.8157	2.0575
NN2	2.2216	2.2216	2.2216

TABLE I

H_∞ SYNTHESIS BENCHMARKS

To illustrate the methodology, we consider the $\mathcal{H}_\infty$ controller synthesis problem for a linear system

$$\begin{cases} \dot{x} = Ax + B_1 w + Bu \\ z = C_1 x + D_{11} w + D_{12} u \\ y = Cx, \end{cases}$$

where x is the state, w the performance input, u the system input, z the performance output and y the physical output. The $\mathcal{H}_\infty$ problem considers determining a static output feedback law $u = Fy$ that optimizes the performance of the system:

$$\begin{aligned} \underset{\gamma, X, F}{\text{minimize}} \quad & \gamma \\ \text{subject to} \quad & \begin{pmatrix} A_F^\intercal X + X A_F & X B_1 & C_F^\intercal \\ B_1^\intercal X & -\gamma I & D_{11}^\intercal \\ C_F & D_{11}^\intercal & -\gamma I \end{pmatrix} \prec 0, \\ & X \succ 0, \end{aligned}$$

where $A_F = A + BFC$ and $C_F = C_1 + D_{12} F C$. In this problem, the bilinearity appears in the term $A_F^\intercal X + X A_F$. By taking F as parameters, we can solve this bilinear matrix inequality problem in two steps as described above.

We solved this problem for a number of test problems in the COMPl$_e$ib collection. The selected problems contained at most 2 parameters. The results are collected in table I and compared to the results reported in [17]. For these small problems, similar or even better performance is achieved compared to literature.

VI. CONCLUSIONS

This paper presents a novel approach for computing approximate solutions to parametric cone programs with a polynomial parameter dependency. The optimizer function is parametrized as a tensor product spline, and in optimizing the coefficients feasibility for all parameter values is guaranteed through the positivity of tensor product B-splines. Relying on duality, bounds on the suboptimality of the approximation are computed, which can be used to locally refine the constraint relaxation and/or the parametrization of the optimizer function. The approach is complemented with an open software tool, and its use is demonstrated in three control-related applications.

Future work will focus on extending the approach to non-hyperrectangular parameter domains and on mitigating the computational complexity for high-dimensional parameter vectors.

ACKNOWLEDGMENT

This work was supported by IWT SBO project MBSE4Mechatronics: Model-based Systems Engineering for Mechatronics; Flanders Make SBO ROCSIS: Robust and Optimal Control of Systems of Interacting Subsystems; KU Leuven-BOF PFV/10/002 Centre of Excellence: Optimization in Engineering (OPTEC); and the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office (DYSCO).

REFERENCES

[1] E. N. Pistikopoulos, L. Dominguez, C. Panos, K. Kouramas, and A. Chinchuluun, "Theoretical and algorithmic advances in multi-parametric programming and control," *Computational Management Science*, vol. 9, no. 2, pp. 183–203, 2012.

[2] A. Alessio and A. Bemporad, "A survey on explicit model predictive control," in *Nonlinear Model Predictive Control*, ser. Lecture Notes in Control and Information Sciences, L. Magni, D. Raimondo, and F. Allgöwer, Eds. Springer Berlin Heidelberg, 2009, vol. 384, pp. 345–369.

[3] N. P. Faísca, V. Dua, B. Rustem, P. Saraiva, and E. N. Pistikopoulos, "Parametric global optimisation for bilevel programming," *Journal of Global Optimization*, vol. 38, no. 4, pp. 609–623, 2007.

[4] A. V. Fiacco, *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*. London, UK: Academic Press, 1983.

[5] T. Johansen and A. Grancharova, "Approximate explicit constrained linear model predictive control via orthogonal search tree," *IEEE Transactions on Automatic Control*, vol. 48, no. 5, pp. 810–815, 2003.

[6] A. Bemporad and C. Filippi, "An algorithm for approximate multiparametric convex programming," *Computational Optimization and Applications*, vol. 35, no. 1, pp. 87–108, 2006.

[7] M. Kvasnica, J. Löfberg, and M. Fikar, "Stabilizing polynomial approximation of explicit MPC," *Automatica*, vol. 47, no. 10, pp. 2292–2297, 2011.

[8] M. Canale, V. Cerone, D. Piga, and D. Regruto, "Fast implementation of model predictive control with guaranteed performance," in *50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, Dec 2011, pp. 3375–3380.

[9] M. Zeilinger, C. Jones, and M. Morari, "Real-time suboptimal model predictive control using a combination of explicit mpc and online optimization," *IEEE Transactions on Automatic Control*, vol. 56, no. 7, pp. 1524–1534, 2011.

[10] Y. Oishi, "Simplified approaches to polynomial design of model predictive controllers," in *2013 IEEE International Conference on Control Applications*, Aug 2013, pp. 960–965.

[11] W. Van Loock, "A matlab toolbox for manipulating and optimizing tensor product splines," https://gitlab.mech.kuleuven.be/meco/splines-m.

[12] C. de Boor, *A Practical Guide to Splines*, ser. Applies Mathematical Sciences. Springer-Verlag New York, Inc., 2001, vol. 27.

[13] L. L. Schumaker, *Spline Functions: Basic Theory*. Cambridge University Press, 2007.

[14] J. Löfberg, "YALMIP : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[15] G. Pipeleers, B. Demeulenaere, J. D. Schutter, and J. Swevers, "Robust high-order repetitive control: Optimal performance trade-offs," *Automatica*, vol. 44, no. 10, pp. 2628–2634, 2008.

[16] D. Henrion and J. Lasserre, "Inner approximations for polynomial matrix inequalities and robust stability regions," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1456–1467, June 2012.

[17] Q. T. Dinh, S. Gumussoy, W. Michiels, and M. Diehl, "Combining convex–concave decompositions and linearization approaches for solving bmis, with application to static output feedback," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1377–1390, 2012.